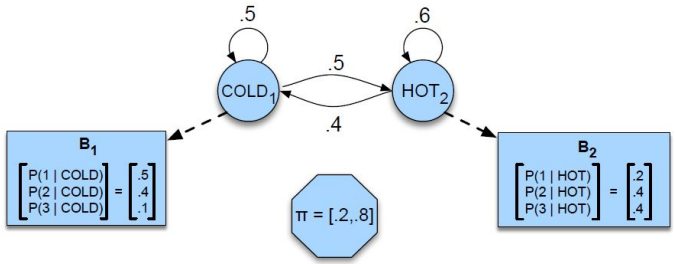
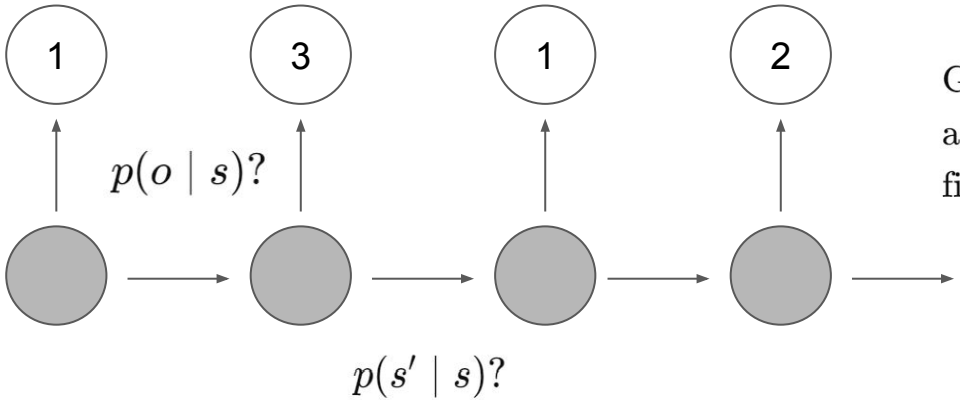
HMM Learning



Problem 3: Learning

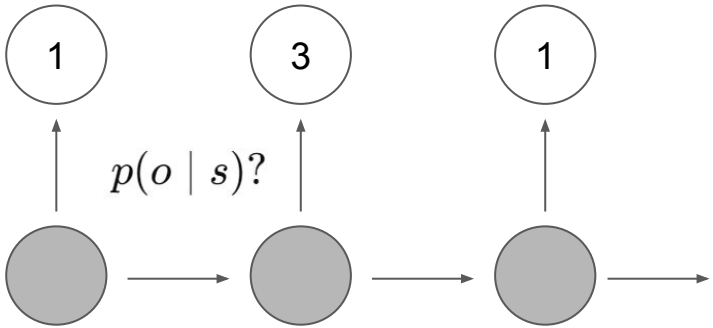




Given $O = \{1, 3, 1, 2, 3, 1, \ldots\}$,
and the set of possible hidden states $\{H, C\}$,
find the T and B matrices.

Observation Probs

Given $O = \{1, 3, 1, 2, 3, 1, \ldots\}$,
and the set of possible hidden states $\{H, C\}$,
find the T and B matrices.



$$p(o^k \mid s^j) = \frac{\text{expected number of times in state } j \text{ and observing } o^k}{\text{expected number of times in state } j}$$

Key: Expected number of times in state j

A Side Problem

You are the admissions coordinator for your department. You are trialing an office assistant robot that your colleague has requested you to test. You have to send offer letters to candidates who recently took the admission test. You ask the robot to do the task.

Unfortunately, there was an error in the visual perception so the robot randomly inserted the letters into envelopes (instead of the correct letter into the correct envelope) and posted them.

Out of the N accepted candidates, how many got their offer letter?

Letter Envelope Matching Problem

$$I_i = \begin{cases} 1 & \text{if the } i\text{th letter is in the correct envelope} \\ 0 & \text{otherwise} \end{cases}$$

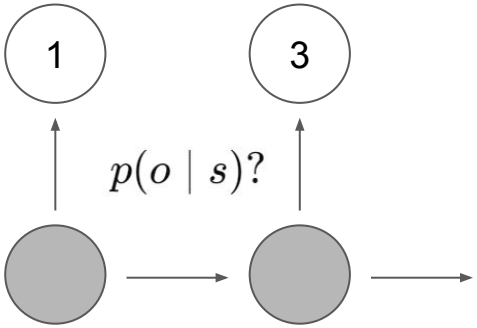
$$S = \sum_{i=1}^{n} I_i$$

Num correct placements

$$\mathbb{E}[S] = \mathbb{E}\left[\sum_{i=1}^{n} I_i\right] := \sum_{i=1}^{n} \mathbb{E}[I_i] \quad = \sum_{i=1}^{n} \frac{1}{n} = 1$$

Expected number of correct letters sent

Learning the Observation Prob.



$$p(o^k \mid s^j) = \frac{\text{expected number of times in state } j \text{ and observing } o^k}{\text{expected number of times in state } j}$$

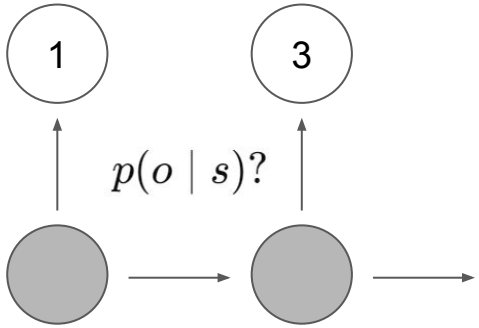
$$I_t = \begin{cases} 1 & \text{if } s_t = j \\ 0 & \text{otherwise} \end{cases}$$

$$S = \sum_{i=1}^{T} I_t$$

Number of times in state j over the sequence of T steps

$$\mathbb{E}[S] = \mathbb{E}\left[\sum_{i=1}^{T} I_t\right] = \sum_{i=1}^{T} \mathbb{E}\left[I_t\right] = \sum_{i=1}^{T} p(s_t = j)$$

Learning the Observation Prob.



$$p(o^k \mid s^j) = \frac{\text{expected number of times in state } j \text{ and observing } o^k}{\text{expected number of times in state } j}$$

$$\gamma_t(j) = p(s_t = j \mid o_{1:T})$$

$$p(o^k \mid s^j) = \frac{\sum_{t=1}^{T} \gamma_t(j) 1(o_t = o_k)}{\sum_{t=1}^{T} \gamma_t(j)}$$

How do we find gamma?

Finding Gamma

$$p(o_k \mid s_j) = \frac{\sum_{t=1}^{T} \gamma_t(j) 1(o_t = o_k)}{\sum_{t=1}^{T} \gamma_t(j)}$$

$$\gamma_t(j) = p(s_t = j \mid o_{1:T})$$

$$= \frac{p(s_t = j, o_{1:T})}{p(o_{1:T})}$$

$$p(o_{1:T}, s_t = j)$$

Numerator

Look familiar?

$$\alpha_t(j) = p(o_{1:t}, s_t = j)$$

Backward Probability

$$p(o_{1:T}, s_t = j)$$

$$p(o_{1:T}, s_t = j) = p(o_{1:t}, s_t = j, o_{t+1:T})$$

$$\beta_t(i) = p(o_{t+1}, o_{t+2}, \ldots, o_T \mid s_t = i)$$

$$p(o_{1:T}, s_t = j) = \alpha_t(j) \cdot \beta_t(j)$$

Why is this true?

Finding Backward Probability

$$\beta_t(i) = p(o_{t+1}, o_{t+2}, \ldots, o_T \mid s_t = i)$$

1. **Initialization:**

$$\beta_T(i) = 1, \quad 1 \leq i \leq N$$

2. **Recursion**

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij}\, b_j(o_{t+1})\, \beta_{t+1}(j), \quad 1 \leq i \leq N, 1 \leq t < T$$

3. **Termination:**

$$P(O|\lambda) = \sum_{j=1}^{N} \pi_j\, b_j(o_1)\, \beta_1(j)$$

Source: SLP, Dan Jurafsky

Back to the Big Picture

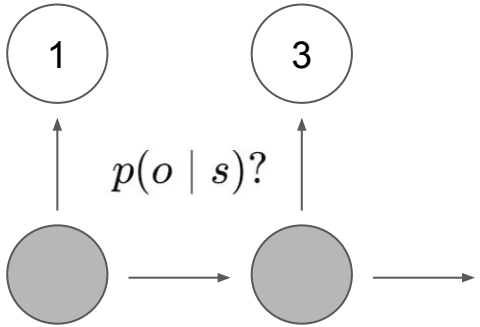
$$p(o_{1:T}, s_t = j) = \alpha_t(j) \cdot \beta_t(j)$$

$$\gamma_t(j) = p(s_t = j \mid o_{1:T})$$

$$= \frac{p(s_t = j, o_{1:T})}{p(o_{1:T})}$$

How do we find this?

$$p(o^k \mid s^j) = \frac{\sum_{t=1}^{T} \gamma_t(j) 1(o_t = o_k)}{\sum_{t=1}^{T} \gamma_t(j)}$$



Consolidation

$$\alpha_t(i) = p(o_{1:t}, s_t = i)$$

$$\beta_t(i) = p(o_{t+1:T} \mid s_t = i)$$

$$\hat{p}(o^k \mid s^j) = \frac{\text{expected number of times in state } j \text{ and observing } o^k}{\text{expected number of times in state } j}$$

$$\gamma_t(j) = p(s_t = j \mid o_{1:T})$$

Expected state occupancy count

$$p(o_{1:T}, s_t = j) = \alpha_t(j) \cdot \beta_t(j)$$

Does this require knowing transition and observation?

Consolidation

$$\alpha_t(i) = p(o_{1:t}, s_t = i)$$

$$\beta_t(i) = p(o_{t+1:T} \mid s_t = i)$$

$$p(o_{1:T}) = \sum_{i=1}^{N} \alpha_T(i)$$

$$\gamma_t(j) = \frac{\alpha_t(j).\beta_t(j)}{\sum_{i=1}^{N} \alpha_T(i)}$$

$$\hat{p}(o^k \mid s^j) = \frac{\sum_{t=1}^{T} \gamma_t(j) 1(o_t = o_k)}{\sum_{t=1}^{T} \gamma_t(j)}$$

Baum-Welch Algorithm

Baum-Welch Algorithm

function FORWARD-BACKWARD(*observations* of len T, *output vocabulary* V, *hidden state set* Q) **returns** $HMM=(A,B)$

initialize A and B

iterate until convergence

E-step

$$\gamma_t(j) = \frac{\alpha_t(j)\beta_t(j)}{\alpha_T(q_F)} \quad \forall\, t \text{ and } j$$

$$\xi_t(i,j) = \frac{\alpha_t(i)\, a_{ij} b_j(o_{t+1})\beta_{t+1}(j)}{\alpha_T(q_F)} \quad \forall\, t,\ i, \text{ and } j$$

M-step

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1}\sum_{k=1}^{N} \xi_t(i,k)}$$

$$\hat{b}_j(v_k) = \frac{\sum_{t=1 s.t.\ O_t=v_k}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)}$$

return A, B

Alert: Use of q for state

Expected state occupancy and state transition counts

Re-estimate the transition and observation probabilities

Source: SLP, Dan Jurafsky

Finding Forward and Backward Probs

Algorithm 3 Baum-Welch Algorithm

Input:
Observation sequence: $O = (O_1, O_2, \ldots, O_T)$
Number of states: N
Number of distinct observation symbols: M
Initial model parameters: transition probabilities A, emission probabilities B, and initial state probabilities π
Output: Updated model parameters: A, B, π
Initialize parameters A, B, π
repeat
    Forward Procedure:
    Initialize α:
    for $i = 1$ to N **do**

$$\alpha_1(i) = \pi_i b_i(O_1)$$

    end for
    Recursively compute α:
    for $t = 2$ to T **do**
        for $j = 1$ to N **do**

$$\alpha_t(j) = \left(\sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} \right) b_j(O_t)$$

        end for
    end for

Initial State Distribution?

    Backward Procedure:
    Initialize β:
    for $i = 1$ to N **do**

$$\beta_T(i) = 1$$

    end for
    Recursively compute β:
    for $t = T - 1$ to 1 **do**
        for $i = 1$ to N **do**

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$$

        end for
    end for

E-Step and M-step: Gamma and Obs Prob

Compute γ:
for $t = 1$ to T **do**
    for $i = 1$ to N **do**

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)}$$

    end for
end for

Re-estimate B:
for $j = 1$ to N **do**
    for $k = 1$ to M **do**

$$b_j(k) = \frac{\sum_{t=1}^{T} \delta(O_t = k)\gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)}$$

where $\delta(\cdot)$ is 1 if the argument is true, 0 otherwise
    end for
end for

Baum-Welch Properties

- Each iteration changes the parameters in a way that is guaranteed to increase the likelihood of the data
- Anytime algorithm: Can stop at any time prior to convergence to get an approximate solution
- Converges to a local maximum